IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

**In re application of:** Sloan et al.

**Application No.** 10/692,361

**Filed:** October 22, 2003

**Confirmation No.** 9370

**For:** HARDWARE-ACCELERATED COMPUTATION OF RADIANCE TRANSFER COEFFICIENTS IN COMPUTER GRAPHICS

FILED VIA EFS ON APRIL 30, 2009

**Examiner:** Said A. Broome

**Art Unit:** 2628

**Attorney Reference No.** 3382-66857-01

FILED VIA EFS
COMMISSIONER FOR PATENTS

## APPEAL BRIEF

Sir:

This brief is in furtherance of the Notice of Appeal filed May 27, 2008. The fee required under 37 CFR 1.17(c) is enclosed.

## I. REAL PARTY IN INTEREST

The real party in interest is Microsoft Corporation, by an assignment from the inventors recorded on October 22, 2003, at Reel 014637, Frame 0155.

## II. RELATED APPEALS AND INTERFERENCES

Currently, there are no other pending appeals or interferences known to appellant, the appellant's legal representatives, or assignees, which will directly affect or be directly affected by or have a bearing on the pending appeal.

## III. STATUS OF CLAIMS

Claims 1, 3, 4, 6-8, 10-12, and 20 remain rejected under 35 U.S.C § 103(a) as unpatentable over *Sloan* et al., *Precomputed Radiance Transfer for Real-Time Rendering in Dynamic, Low-Frequency Lighting Environments* (hereinafter "*Sloan*") in view of *Burke*, U.S. Patent Publication No. 2003/0063096 (hereinafter "*Burke*") in further view of *Purcell* et al., *Ray Tracing on Programmable Graphics Hardware* (hereinafter "*Purcell*").

Claims 2, 5, 12, 13, and 15-19 remain rejected under 35 U.S.C § 103(a) as unpatentable over *Sloan* in view of *Morioka* et al., U.S. Patent No. 6,333,742 (hereinafter "*Morioka*") in further view of *Burke* and in further view of *Purcell*.

Claim 9 remains rejected under 35 U.S.C § 103(a) as unpatentable over *Sloan* in view of *Burke*, in further view of *Purcell*, and in further view of *Arvo* et al., *Monte Carlo Ray Tracing* (hereinafter "*Arvo*").

Claim 14 remains rejected under 35 U.S.C § 103(a) as unpatentable over *Sloan* in view of *Morioka*, in further view of *Burke*, in further view of *Purcell*, and in further view of *Airey* et al., U.S. Patent No. 6,650,327 (hereinafter "*Airey*").

All the above pending claims (i.e., claims 1-20) are appealed in conjunction with which the subject Appeal Brief is being filed.

## IV. STATUS OF AMENDMENTS

A qualifying amendment was filed on September 10, 2007 and was entered. Thus, for the purpose of Appeal the claims will be presented as they appeared after the entry of the amendment filed on September 10, 2007.

## V. SUMMARY OF CLAIMED SUBJECT MATTER

The claims on appeal relate to computer graphics image rendering, and more particularly to innovations that permit a portion of the rendering process (more specifically, the radiance transfer computation) to be more suitable for execution on programmable graphics processing unit (GPU) hardware. See Specification at page 3, line 14-17. The radiance transfer computation of the rendering process calculates a radiance transfer quantity for each of a plurality of sampled points on the surface of an object to be rendered in the computer graphics image scene. See Specification at page 5, line 5 through page 7, line 10 (section entitled "Precomputed Radiance Transfer Overview").

According to the independent claim 1 on appeal (which recites, "creating an object positions texture...," and "creating an object normals texture"), the radiance transfer computation is performed with texture-based operations using a graphics processing unit, which

operate on textures respectively containing positions and normals of the sampled points mapped into a texture space. See Specification at page 9, lines 4-10; Figure 5, item 510; and Figure 7, variable s0, s1 in listing 700.

Similarly, according to the independent claim 2 on appeal (which recites, "wherein the radiance transfer coefficients processing program... creates an object positions texture..., and creates an object normals texture"), the radiance transfer computation is performed with texture-based operations using a graphics processing unit, which operate on textures respectively containing positions and normals of the sampled points mapped into a texture space. See Specification at page 9, lines 4-10; Figure 5, item 510; and Figure 7, variable s0, s1 in listing 700.

Similarly, according to the independent claim 3 on appeal (which recites, "code means executable on a computer for creating an object positions texture...," and "code means executable on a computer for creating an object normals texture"), the radiance transfer computation is performed with texture-based operations using a graphics processing unit, which operate on textures respectively containing positions and normals of the sampled points mapped into a texture space. See Specification at page 9, lines 4-10; Figure 5, item 510; and Figure 7, variable s0, s1 in listing 700.

Further according to the independent claim 1 (which recites, "iteratively, for each of a set of directions sampled about the object," performing various texture-based operations ("determining cosine terms," "determining shadowing," and "determining radiance transfer contribution") over a set of sampled points), the radiance transfer computation has an inner loop that iterates over the sampled points, and an outer loop iterating over directions. See Specification at page 8, line 1 through page 9, line 4; and Figure 4.

Similarly, according to the independent claim 2 (which recites, "wherein the at least one pixel shader executing on the graphics processing unit performs texture operations that iteratively, for each of a set of directions sampled about the object," performs various texture-based operations ("determine cosine terms," "determine shadowing," and "determine radiance transfer contribution") over a set of sampled points), the radiance transfer computation has an inner loop that iterates over the sampled points, and an outer loop iterating over directions. See Specification at page 8, line 1 through page 9, line 4; and Figure 4.

Similarly, according to the independent claim 3 (which recites, "code means executable on the graphics accelerating hardware of the computer to perform texture operations that iteratively, for each of a set of directions sampled about the object," performs various texture-based operations ("determine cosine terms," "determine shadowing," and "determine radiance transfer contribution") over a set of sampled points), the radiance transfer computation has an inner loop that iterates over the sampled points, and an outer loop iterating over directions. See Specification at page 8, line 1 through page 9, line 4; and Figure 4.

Further details of an embodiment implementing the claimed invention are disclosed in the Specification at page 4, line 25 through page 15, line 8; and Figures 4-9.

## VI. GROUNDS OF REJECTION TO BE REVIEWED ON APPEAL

The grounds of the following rejections are to be reviewed on appeal:

1. The rejection of claims 1, 3, 4, 6-8, 10-12, and 20 under 35 U.S.C § 103(a) as unpatentable over *Sloan*, in view of *Burke*, in further view of *Purcell*.
2. The rejection of claims 2, 5, 12, 13, and 15-19 under 35 U.S.C § 103(a) as unpatentable over *Sloan* in view of *Morioka*, in further view of Burke and in further view of Purcell..

3. The rejection of claim 9 under 35 U.S.C § 103(a) as unpatentable over *Sloan* in view of *Burke*, in further view of *Purcell*, and in further view of *Arvo*.

4. The rejection of claim 14 under 35 U.S.C § 103(a) as unpatentable over *Sloan* in view of *Morioka*, in further view of *Burke*, in further view of *Purcell*, and in further view of *Airey*.

## VII. ARGUMENT

A proper obviousness rejection requires that a reference or a combination of references teach or suggest each and every claim feature. MPEP § 2142. However, "when prior art teaches away from combining certain known elements, discovery of a successful means of combining them is more likely to be nonobvious." *See Id.* and *KSR Int'l Co. v. Teleflex Inc.*, 550 U.S. ___, 82 U.S.P.Q.2d 1385, 1395 (2007). "A reference may be said to teach away when a person of ordinary skill, upon reading the reference, would be discouraged from following the path set out in the reference, or would be led in a direction divergent from the path that was taken by the applicant." *In re Icon Health and Fitness, Inc.*, ___ F.3d ___, 83 U.S.P.Q.2d 1746, 1751 (Fed. Cir. 2007) (quoting *In re Gurley*, 27 F.3d 551, 553, 31 U.S.P.Q.2d 1130 (Fed. Cir. 1994)). Also, "a proposed modification [is] inappropriate for an obviousness inquiry when the modification render[s] the prior art reference inoperable for its intended purpose." *In re Fritch*, 972 F.2d 1260, 23 U.S.P.Q.2d 1780, 1783 n.12 (Fed. Cir. 1992). The cited references in the obviousness rejections do not render the appealed claims obvious under these tests.

A. Claims 1-20 would not have been obvious over the asserted art.

The asserted art fails to teach or suggest at least the following aspects of the claimed invention: (1) a computation of the radiance transfer for each sampled point, which computation uses an outer loop iterating over directions and inner loop iterating over points; and (2) texture operations on textures containing data values representing normals and positions of sample points mapped into a texture space.

The asserted art lacks computation of radiance transfer for sampled points with outer loop iterating over directions, and inner loop iterating over points.

The calculation of radiance transfer for a set of sampled points using an outer loop iterating over directions, and texture-based operations on sets of sampled points inside the loop is not taught or suggested by this art. The independent claims 1, 2 and 3 each recites language relating to a radiance transfer computation with an outer loop iterating over directions, and inner loop iterating over points. For example, claim 1 recites, "iteratively, for each of a set of directions sampled about the object," performing various texture-based operations ("determining cosine terms," "determining shadowing," and "determining radiance transfer contribution") over a set of sampled points. The independent claims 1, 2 and 3 also recite language relating to the calculation yielding a radiance transfer value per sampled point. In particular, claim 1 recites the claimed method "produc[es] a radiance transfer value for each of the sampled points from the accumulated radiance transfer contributions for the iterated directions at the respective sampled points." Claims 2 and 3 recite like language.

Examiner Broome asserts that it would have been obvious to reverse the inner and outer loops of the prior Sloan radiance computation illustrated in Figure 3 of the Specification, in view of the description by Purcell of an optimization to minimize the total number of passes by a ray

tracer. See Office Action mailed 11/26/07 at page 4, line 12 through page 5, line 7. Appellants respectfully disagree, and submit that the proposed modification fails to establish a prima facie case of obviousness because (1) Purcell fails to suggest the proposed modification of an outer loop iterating over directions and inner loop iterating over points, (2) the proposed modification would render the prior Sloan radiance transfer computation inoperable for its intended purpose, and (3) the Examiner fails to articulate adequate rationale how Purcell would have led the person of ordinary skill in the art to make the proposed modification.

Purcell fails to suggest the proposed modification of an outer loops iterating over directions and inner loop iterating over points. First, Examiner Broome alleges that the description by Purcell (at section 3.2, ¶ 4, lines 1-5 and 10-14, which reads "*we present an optimization to minimize the total number of passes... There are various strategies for nesting these loops... For graphics hardware... The following is a more efficient algorithm...*") suggests the proposed modification of reversing loop parameters to iterate over directions in an outer loop and points in an inner loop. Appellants disagree. When Purcell speaks of there being "various strategies for nesting these loops" (Purcell, s. 3.2, ¶ 4), he is not referring to iterating over points versus iterating over directions. Actually, Purcell is referring to traversal and intersection kernels of the Delany ray tracer. (Purcell, s. 3.2, ¶ 4) As can be seen from the pseudo-code listing in Purcell, s. 3.2 (right side of page), both the traversal and intersection kernels operate on rays ("traverse(ray)" and "intersect(ray)"). There is nothing to indicate that either kernel iterates over points.

Further, it would be apparent to one of ordinary skill in the art that these two kernels do not form inner and outer loops. Rather, the traverse and intersect kernel are contained in an "if... else..." statement. See Purcell, s. 3.2 (right side of page). The traverse and intersect kernel

"loops" therefore are alternative execution paths in Purcell's optimized ray tracer, and are not structured as inner and outer loops of the ray tracer.

Moreover, the optimized ray tracer described by Purcell does not appear to involve iterating over points at all. Instead, the ray tracer merely iterates over a selection of rays projected from the eye (a single point) into the scene. See Purcell, s. 3.2 (reference in pseudo-code at right side of page to "generate eye ray," which "ray" appears to be the parameter over which the while loop is iterated). Presumably, if there were more than one observation point or eye for the scene, Purcell's optimized ray tracer would be repeated for each eye. If so, then Purcell also describes an algorithm that iterates over directions (e.g., the "rays") in an inner loop, and points (the single eye) in an outer loop. This would be no different than the prior Sloan radiance transfer process (illustrated in Figure 3 of Appellants' patent application, and described at page 7, line 12 through page 8, line 20.) Purcell therefore at most suggests there are various strategies for nesting of traverse and intersect kernels of the Delany ray tracer, and lacks any teaching or suggestion of iterating over directions as an outer loop and points as an inner loop of a ray tracer. Purcell simply fails to teach or suggest the proposed modification to Sloan.

The proposed modification would render the prior Sloan method inoperable for its intended purpose. Second, the Examiner's proposed modification (the reversal of the parameters iterated over by outer and inner loops of the prior art calculation illustrated in Figure 3 of the Specification) would not result in the claimed method that produces radiance transfer at each point which is the accumulated radiance transfer contribution over all directions for that point. For example, the prior art computation as shown in Figure 3 of the present application is as follows:

```
For each point P
     Accum = 0
     For each direction D
          Hn = dot(D,N)
          if (Hn < 0) continue;
          if (RayDoesNotIntersect(P,D))
               Accum += B(D)*Hn
     End For
     T = Accum/Norm
End For
```

If one were to modify the process by switching the values iterated by inner and outer loops as proposed by the Examiner, the resulting calculation would be:

```
For each direction D
     Accum = 0
     For each point P
          Hn = dot(D,N)
          if (Hn < 0) continue;
          if (RayDoesNotIntersect(P,D))
               Accum += B(D)*Hn
     End For
     T = Accum/Norm
End For
```

This "reversed loops" calculation produces a transfer value per direction that is the accumulated contribution from all points for that direction. The transfer per direction produced from the reversed calculation is not the same as the radiance transfer per point that is the accumulated contribution from all directions as recited in claims 1-3. (Claim 1 recites, "producing a radiance transfer value for each of the sampled points from the accumulated radiance transfer contributions for the iterated directions at the respective sampled points.") In other words, the claimed method produces a radiance transfer value for each point that integrates the radiance contributions from all iterated directions. The calculation as per the proposed modification yields a radiance transfer for each direction that integrates contributions over all sampled points of the object surface. These two quantities are very different. The "radiance transfer per point" quantity of the claimed method is suitable for use in producing computer graphics images with the pre-computed radiance transfer rendering technique described by Sloan. The "radiance transfer per direction" quantity produced by the proposed modification is not suitable to Sloan's pre-computed radiance transfer rendering technique. So, the proposed modification merely reversing what value is iterated in the outer and inner loops of the prior Sloan calculation actually does not produce the radiance transfer at each point for the iterated directions, as recited in Appellants' claims. Moreover, the proposed modification to the prior Sloan computation would render it inoperable for its intended purpose – rendering of computer graphics images with the rendering technique.

The Examiner fails to articulate adequate rationale how Purcell would have led the person of ordinary skill in the art to make the proposed modification. Third, the cited art would not motivate reversing the values iterated by outer and inner loops of the prior Sloan radiance transfer computation illustrated in Figure 3 of the specification. Examiner Broome alleges that

Purcell's description of optimizing a ray tracing procedure "to minimize the total number of passes" would motivate modifying the prior Sloan radiance transfer computation illustrated in Figure 3 of the Specification. See Office Action mailed 11/26/07 at page 4, line 12 through page 5, line 7. Appellants respectfully disagree.

Unlike the ray tracing process described in Purcell, switching the values iterated by outer and inner loops of the radiance transfer process illustrated in Figure 3 of the Specification does not permit optimizing to "minimize the total number of passes" as achieved by Purcell. In Purcell's ray tracer, the single "while" loop iterates through "rays." *See*, Purcell at section 3.2, ¶ 4. This while loop includes a test of whether any rays are "active." *See*, Purcell at section 3.2, ¶ 5. The "inactive" rays have "either hit triangles or traversed the entire grid," and therefore do not require a further traverse or intersect pass. *See*, Purcell at section 3.2, ¶ 5. The simple procedure, which Purcell lists in section 3.2 between ¶¶ 4 and 5, runs the intersection test if any rays require intersection. *See*, Purcell at section 3.2, ¶ 5. Purcell describes that this procedure can be optimized by deciding to perform "intersection" passes only when 20% of the rays require intersection tests. *See*, Purcell at section 3.2, ¶ 5. By contrast, the proposed modification to the prior Sloan radiance transfer computation illustrated in Figure 3 of the Specification would not eliminate any inner loop iteration or "pass." Accordingly, the motivation for Purcell's optimization of nesting loops to eliminate "intersection kernel" passes does not apply to the radiance transfer computation in Figure 3.

In the particular context of the prior Sloan radiance transfer computation, the Sloan method includes not only contributions from direct radiation, but also includes contributions from interreflections. See Specification at page 7, lines 12-22. Because it includes interreflection transfer from occluded directions, the Sloan radiance transfer computation does

not permit skipping occluded directions. Purcell's optimization that skips passes for occluded directions would be contrary to the inclusion of interreflection contributions by Sloan radiance transfer computation, and making the proposed modification for the purpose of eliminating passes for occluded directions would render the method inoperable for its intended purpose of computing radiance transfer including interreflection.

In fact, in the claimed method recited in claims 1-3, the texture-based operations of "determining cosine terms," "determining shadowing," and "determining radiance transfer contribution" are recited in the claims as being preformed "for each of a set of directions." The choice of value iterated by the outer loops does not permit eliminating "passes" or iterations of the inner loop. Rather, all these steps are performed for each direction iterated in the outer loop. The motivation for the optimization described by Purcell therefore would not have led to changing which parameter is iterated by the outer loop of the radiance transfer method illustrated in Figure 3.

In describing the proposed modification allegedly motivated by Purcell, Examiner Broome quotes the following statement in Purcell: "There are various strategies for nesting these loops." However, in using the phrase "these loops," Purcell is referring specifically to the traverse and intersect kernels of the ray tracing algorithm that is the subject of Section 3.2 of Purcell's paper. The statement observes that the kernels of the subject ray tracing algorithm are susceptible to various loop nesting strategies. However, the statement does not appear to have been intended by Purcell as summarizing some general principle that various nesting strategies would exist for all algorithms, and would not be taken as such by one of ordinary skill in the art at the time of the invention. The statement says nothing about whether other nesting strategies

were known to exist for the outer/inner loop parameters of the prior Sloan radiance transfer computation method shown in Figure 3 of the Specification.

While Appellants have shown that Purcell's optimization of eliminating intersection passes for occluded directions is inapplicable to the prior Sloan radiance transfer method and therefore could not have lead one of ordinary skill in the art to make the proposed modification, Examiner Broome in the Office Action mailed 11/27/07 at page 23, lines 1-17 (more particularly at lines 6-9 and 17) now also points to a statement in the detailed description of Appellants' patent application (Specification, page 8, lines 26-28) as allegedly providing motivation to one of ordinary skill in the art at the time of the invention to make the proposed modification. Appellants respectfully submit that Examiner's Broome's reliance on this statement fails to establish a prima facie case of obviousness of the appealed claims because (1) the statement merely touts a benefit of a disclosed embodiment and not the proposed modification itself, and (2) such reliance amounts to an impermissible exercise of hindsight reasoning.

First, the statement in question compares a detailed embodiment of the claimed invention ("hardware accelerated PRT preprocess 400") to the prior Sloan radiance transfer method, and touts the detailed embodiment is "more suitable for GPU execution" compared to the prior Sloan radiance transfer method. The statement indicates a reason for this benefit is that the order of the inner- and outer-loops is reversed. While the statement lacks the qualification that the benefit is due only "in part" to this difference, it also does not say that the benefit is due solely to this distinction between the detailed embodiment and the prior Sloan radiance transfer method. In fact, the detailed embodiment differs in a number of other ways from the prior Sloan radiance transfer method. Further, the mere reversal of the values iterated in the inner and outer loops alone would render the process inoperable for the purpose of producing a radiance transfer

quantity for each sample point over the iterated directions, as discussed in detail above. Instead, it is not only the reversal of inner/outer loops but the combination with other aspects of the detailed embodiment that produce this benefit. Accordingly, it would be taking the statement out of its intended context to read the statement as a general assertion that mere reversal of the inner/outer loop parameters alone yields a process more suitable for execution on a GPU.

Second, Examiner Broome's reliance on a statement in the Specification to provide motivation for the proposed modification amounts to an impermissible exercise of hindsight reasoning. The statement relied upon by the Examiner appears in the detailed description of Appellants' patent application. The application was first published on April 28, 2005, clearly many months after the filing of the application and constructive date of the invention. How then would this detailed description of an embodiment of the present invention in Appellants' patent application have led the person of ordinary skill in the art at the time of the invention toward the proposed modification? In making this statement comparing the detailed embodiment to the prior Sloan radiance transfer method, Appellant did not indicate this benefit of Appellants' detailed embodiment was common knowledge in the art. Moreover, Appellant was most certainly not declaring some kind of basic principal that mere reversal of loop parameters makes a process more suitable for GPU execution, nor declaring such to be common knowledge at the time of invention. Therefore, one of ordinary skill in the art could not have been led to reverse the inner/outer loops of the prior Sloan radiance transfer method for the purpose of making the process more suitable for GPU execution.

The asserted art lacks texture operations on textures containing data values representing normals and positions of sample points mapped into a texture space.

The texture operations on textures containing data values representing normals and positions of sample points mapped into a texture space also is not taught or suggested by the asserted art. The independent claims 1-3 each recite language relating to performing texture operations on textures containing data values representing normals and positions of sample points mapped into a texture space. For example, claim 1 recites "creating an object positions texture containing a set of data values representing positions of a set of points sampled over the object mapped into a texture space," and "creating an object normals texture containing a set of data values representing normals of the set of sampled points mapped into the texture space." Claim 1 then recites various steps performed as texture-based operations involving these textures. Claims 2 and 3 express like language.

Examiner Broome recognizes that "Sloan fails to teach creating an object positions and normal texture," and asserts "Burke teaches creating an object positions texture representing positions of a set of points sample over the object... and object normals texture representing normals..." at paragraph 0035, lines 4-10 of Burke. See Office Action mailed 11/27/07 at page 4, lines 7-12. Appellants respectfully disagree, and submit that Burke lacks any teaching or suggestion to put position and normal data values that are mapped into texture space into textures for processing using texture-based operations of a graphics processing unit, and then performing texture-based operations using a graphics processing unit to process such position and normal data in the textures.

At the cited paragraph 0035, lines 4-10, Burke indicates each sampled point is represented by nine data values, position (X, Y, Z), color (R, G, B), and normal (I, J, K). Burke

also describes a model data creator creates a data structure for this nine-dimension data values of the sampled points, and indicates the data structure is "in the form of a network where each point has four pointers [indicating next points in U, V directions]." *See*, Burke at paragraph 0036. Appellants submit that this network of pointers is not a texture structure that is processed using texture-based operations of a graphics processing unit.

Regardless of whether Burke's "network of pointers" constitutes a texture that is processed using texture-based operations of a graphics processing unit, Burke also indicates that the data values of Burke's data structure are not mapped to the texture space. Burke at paragraph 9 states, "The present invention re-samples the data in each group as a vector-valued field in a parametric U and V grid space. This space is not the same as the texture UV space." Burke therefore fails to suggest textures containing data values representing positions or normals of sampled points mapped into the texture space, as claimed.

For the reasons stated above, the asserted art fails to teach or suggest the limitations recited in the independent claims 1-3, and their dependent claims 4-20. Appellants therefore respectfully request that the rejection of these claims be reversed.

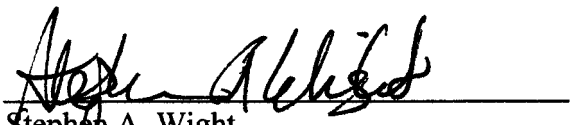
## VIII. CONCLUSION

In light of the arguments presented above the rejection of claims 1-20 should be reversed and all claims passed to issue.

Respectfully submitted,

KLARQUIST SPARKMAN, LLP

One World Trade Center, Suite 1600
121 S.W. Salmon Street
Portland, Oregon 97204
Telephone: (503) 595-5300
Facsimile: (503) 595-5301

By 
Stephen A. Wight
Registration No. 37,759

# APPENDIX

## CLAIMS AS PENDING

1. (Rejected) A method of producing radiance transfer coefficients for a set of points sampled over a modeled object for rendering images of the object on a computer having a graphics processing unit for performing operations over sets of data values contained in textures, the method comprising:

creating an object positions texture containing a set of data values representing positions of a set of points sampled over the object mapped into a texture space;

creating an object normals texture containing a set of data values representing normals of the set of sampled points mapped into the texture space;

iteratively, for each of a set of directions sampled about the object,

rendering the object from the direction to produce a shadow buffer representing depth from the object in the direction for the set of points;

as a texture-based operation using the graphics processing unit, determining cosine terms of the set of sampled points for the currently iterated direction based on the normals represented in the object normals texture and currently iterated direction;

as a texture-based operation using the graphics processing unit, determining shadowing of the set of sampled points for the currently iterated direction based on the depths represented in the shadow buffer and positions represented in the object positions texture;

as a texture-based operation using the graphics processing unit, determining radiance transfer contribution of the set of sampled points for the currently iterated direction based on the determined cosine terms and shadowing; and

accumulating the radiance transfer contributions of the set of sampled points for the currently iterated direction with that of previously iterated directions;

producing a radiance transfer value for each of the sampled points from the accumulated radiance transfer contributions for the iterated directions at the respective sampled points;

rendering an image of the object in a lighting environment based on the accumulated radiance transfer contributions; and

presenting the image.

2. (Rejected) A computer system for hardware-accelerated processing of a radiance transfer coefficients computation for a set of points sampled over a modeled object for use in rendering images of the object, the computer system comprising:

a memory for storing program code of at least one pixel shader and a radiance transfer coefficients processing program;

a central processing unit operating to execute the radiance transfer coefficients processing program;

a graphics processing unit programmable by and operating to execute the at least one pixel shader;

wherein the radiance transfer coefficients processing program executing on the central processing unit creates an object positions texture that contains data values representing positions of a set of points sampled over the object mapped into a texture space, and creates an object

normals texture that contains data values representing normals of the set of sampled points mapped into the texture space;

wherein the at least one pixel shader executing on the graphics processing unit performs texture operations that iteratively, for each of a set of directions sampled about the object,

render the object from the direction to produce a shadow buffer representing depth from the object in the direction for the set of points;

determine cosine terms of the set of sampled points for the currently iterated direction based on the normals represented in the object normals texture and currently iterated direction;

determine shadowing of the set of sampled points for the currently iterated direction based on the depths represented in the shadow buffer and positions represented in the object positions texture;

determine radiance transfer contribution of the set of sampled points for the currently iterated direction based on the determined cosine terms and shadowing; and

accumulate the radiance transfer contributions of the set of sampled points for the currently iterated direction with that of previously iterated directions; and

wherein the graphics processing unit produces a radiance transfer value for each of the sampled points from the accumulated radiance transfer contributions for the iterated directions at the respective sampled points.

3. (Rejected) Computer-readable media having stored thereon programming code executable at least in part on graphics accelerating hardware on a computer to perform

processing of a radiance transfer coefficients computation for a set of points sampled over a modeled object for use in rendering images of the object, the programming code comprising:

code means executable on a computer for creating an object positions texture that contains data values representing positions of a set of points sampled over the object mapped into a texture space;

code means executable on a computer for creating an object normals texture that contains data values representing normals of the set of sampled points mapped into the texture space;

code means executable on the graphics accelerating hardware of the computer to perform texture-based operations that iteratively, for each of a set of directions sampled about the object,

render the object from the direction to produce a shadow buffer representing depth from the object in the direction for the set of points;

determine cosine terms of the set of sampled points for the currently iterated direction based on the normals represented in the object normals texture and currently iterated direction;

determine shadowing of the set of sampled points for the currently iterated direction based on the depths represented in the shadow buffer and positions represented in the object positions texture;

determine radiance transfer contribution of the set of sampled points for the currently iterated direction based on the determined cosine terms and shadowing; and

accumulate the radiance transfer contributions of the set of sampled points for the currently iterated direction with that of previously iterated directions; and

code means executable on the computer to produce a radiance transfer value for each of the sampled points from the accumulated radiance transfer contributions for the iterated directions at the respective sampled points.

4. (Rejected) The method of claim 1 wherein the texture-based operations for determining cosine terms, determining shadowing, determining radiance transfer contributions, and said accumulating radiance transfer contributions form an inner computational loop that iterates over the sampled points, and wherein an outer computational loop iteratively repeats the inner computational loop over the set of sampled directions.

5. (Rejected) The computer system of claim 2 wherein the texture operations that render, determine cosine terms, determine shadowing, determine and accumulate radiance transfer contributions form an inner computational loop that iterates over the sampled points, and wherein an outer computational loop iteratively repeats the inner computational loop over the set of sampled directions.

6. (Rejected) The method of claim 1 wherein the object positions texture contains an arrangement of data values representing the position of each of the sampled points mapped into the texture space.

7. (Rejected) The method of claim 4 wherein the object positions texture is stored in an RGB component format.

8. (Rejected) The method of claim 1 wherein the object normals texture contains an arrangement of data values representing the surface normal at each of the sampled points mapped into the texture space.

9. (Rejected) The method of claim 1 wherein the set of directions are generated as uniformly distributed points on a unit sphere based on a mapping from the unit square to the sphere and jittered sampling.

10. (Rejected) The method of claim 1 wherein said determining cosine terms, determining shadowing, determining radiance transfer contribution, and accumulating the radiance transfer contributions are performed using a pixel shader executed on a programmable graphics processing unit.

11. (Rejected) The method of claim 1 wherein said rendering the object from the direction comprises rendering the object as an orthographic camera projection whose view direction is set to the current direction.

12. (Rejected) The method of claim 11 wherein said determining shadowing comprises for each of the sampled points:

computing depth of a current sampled point based on the current sampled point's position as represented in the object positions texture;

comparing the computed depth of the current sampled point to an object depth from the current direction as represented in the shadow buffer to determine visibility of the current sampled point in the current direction.

13. (Rejected) The computer system of claim 2 wherein the object positions texture contains an arrangement of data values representing the position of each of the sampled points mapped into the texture space.

14. (Rejected) The computer system of claim 2 wherein the object positions texture is stored in a floating point number format.

15. (Rejected) The computer system of claim 2 wherein the object normals texture contains an arrangement of data values representing the surface normal at each of the sampled points mapped into the texture space.

16. (Rejected) The computer system of claim 2 wherein the set of directions are to uniformly distributed points on a unit sphere.

17. (Rejected) The computer system of claim 2 wherein said rendering the object from the direction comprises rendering the object as an orthographic camera projection whose view direction is set to the current direction.

18. (Rejected) The computer system of claim 17 wherein said determining shadowing comprises for each of the sampled points:

computing depth of a current sampled point based on the current sampled point's position as represented in the object positions texture;

comparing the computed depth of the current sampled point to an object depth from the current direction as represented in the shadow buffer to determine visibility of the current sampled point in the current direction.

19. (Rejected) The computer-readable media of claim 3 wherein said code means executable on the graphics accelerating hardware of the computer to perform texture-based operations is a pixel shader executable on a programmable graphics processing unit.

20. (Rejected) The computer-readable media of claim 3 wherein said code means executable on the graphics accelerating hardware comprises an inner computational loop that iterate over the sample points as the texture operations that determine cosine terms, determine shadowing, determine and accumulate radiance transfer contributions form an inner computational loop; and an outer computational loop that iterates the inner computational loop over the set of sampled directions.

# EVIDENCE

None

# RELATED CASES

None